Previous Doc Next Doc Go to Doc#
First Hit

Generate Collection

L17: Entry 1 of 2 File: PGPB Sep 12, 2002

DOCUMENT-IDENTIFIER: US 20020126685 A1
TITLE: Time division multiplexing over broadband modulation method and apparatus

Application Filing Date:
20010312

Detail Description Paragraph:
[0053] FIG. 4 is a functional block diagram of an exemplary embodiment of a CIM 305. The CIM 305 forwards IP packets and performs packet framing and channelization. In addition, the CIM 305 performs the associated digital signal and RF processing for transmission over the network architecture. Each CIM 305 includes a cell processing engine 405 that interfaces the switch 303 via a switch interface 409. The cell processing engine 405 may include supporting internal or external memory for table-lookups, queued data payload buffer descriptors and data payload buffer storage. Such memory may include any combination of read only memory (ROM) or random access memory (RAM) devices. The cell processing engine 405 processes each packet transferred between the network interface via the splitter/RX 207 and the switch interface 409. The cell processing engine 405 functionality includes IP forwarding, link layer framing and physical layer encoding for transmission to the combiner/TX 205 or to switch interface 409 for transmission to the switch 303. In addition, the cell processing engine 405 performs physical and link layer framing.

Detail Description Paragraph:
[0061] FIG. 6A is a block diagram illustrating IP packet decapsulation and cell encapsulation for downstream transmission by the cell processing engine 405. As further described below, synchronous, byte-oriented processing utilizes packet and cell headers to allow variable-length IP packets to be transported across the network as a series of payload cells. These packet and cell headers provide the destination with enough information to reassemble the individual cells back into the original IP packets to decode the message. The adaptation and convergence procedures, described further below, also perform null packet generation and added error protection. It is understood that although the present invention is illustrated with IP packets, the present invention applies to any type of digital information, including various types of packetized information and data packets.

Detail Description Paragraph:
[0068] FIG. 6B is a block diagram illustrating CCP and PAP header agreement between successive CCP cells 651 and 653. To ensure that the CPE of the subscriber destination 109 can reliably reassemble IP packets from a series of individual CCP cells, the CCP verifies that the pointer offset values and the previous PAP header's length field are in agreement. The first CCP cell 651 is followed by a subsequent CCP cell 653, each including respective CCP headers 655 and 657. The CCP cells 651 and 653 are not necessarily consecutive, in which case intermediate CCP cells include a CCP header with the maximum value. The first CCP cell 651 includes a PAP header 659 and a corresponding first portion of an IP packet 1. The CCP header 655 includes a pointer offset value indicating the position of the PAP header 659 within the CCP cell 651. The PAP header 659 includes the length field 621 defining the length of IP packet 1, and therefore indicates the location of a subsequent PAP header 661 within the subsequent CCP cell 653. The PAP header 661 is

located at the beginning of the next subsequent IP packet 2. The CCP header 657 includes a pointer offset value indicating the position of the PAP header 661 within the CCP cell 653. Thus, the CCP verifies that the CCP header 657 and the PAP header 659 are in agreement as to the location of the next PAP header 661.

Detail Description Paragraph:
[0080] Following the encoding process, a convolutional interleaving scheme is applied resulting in interleaved frames (not shown). In one embodiment, the resulting interleaved frames are composed of overlapping error-protected packets that are delimited by sync bytes to preserve a periodicity of 204 bytes. The frames may be interleaved in accordance with the ITU J.83 specification and will not be further described. The interleaved frames are then modulated, such as according to QAM-256 modulation as provided in the ITU J.83 specification. The QAM process adapts the synchronous, scrambled bit-stream for transmission over a channel as RF output. The QAM process blocks together bits from the data stream and then maps them into codewords using either Gray-codes or differential codes. The QAM process then converts the resulting digital codewords into an analog waveform based on a constellation diagram of combinations of amplitudes and phases, where each unique bit sequence corresponds to a point in the constellation.

Previous Doc Next Doc Go to Doc#

Previous Doc Next Doc Go to Doc#
First Hit Fwd Refs

Generate Collection

L17: Entry 2 of 2 File: USPT Jul 13, 2004

DOCUMENT-IDENTIFIER: US 6763025 B2
TITLE: Time division multiplexing over broadband modulation method and apparatus

Application Filing Date (1):
20010312

Detailed Description Text (25):
FIG. 4 is a functional block diagram of an exemplary embodiment of a CIM 305. The CIM 305 forwards IP packets and performs packet framing and channelization. In addition, the CIM 305 performs the associated digital signal and RF processing for transmission over the network architecture. Each CIM 305 includes a cell processing engine 405 that interfaces the switch 303 via a switch interface 409. The cell processing engine 405 may include supporting internal or external memory for table-lookups, queued data payload buffer descriptors and data payload buffer storage. Such memory may include any combination of read only memory (ROM) or random access memory (RAM) devices. The cell processing engine 405 processes each packet transferred between the network interface via the splitter/RX 207 and the switch interface 409. The cell processing engine 405 functionality includes IP forwarding, link layer framing and physical layer encoding for transmission to the combiner/TX 205 or to switch interface 409 for transmission to the switch 303. In addition, the cell processing engine 405 performs physical and link layer framing.

Detailed Description Text (33):
FIG. 6A is a block diagram illustrating IP packet decapsulation and cell encapsulation for downstream transmission by the cell processing engine 405. As further described below, synchronous, byte-oriented processing utilizes packet and cell headers to allow variable-length IP packets to be transported across the network as a series of payload cells. These packet and cell headers provide the destination with enough information to reassemble the individual cells back into the original IP packets to decode the message. The adaptation and convergence procedures, described further below, also perform null packet generation and added error protection. It is understood that although the present invention is illustrated with IP packets, the present invention applies to any type of digital information, including various types of packetized information and data packets.

Detailed Description Text (40):
FIG. 6B is a block diagram illustrating CCP and PAP header agreement between successive CCP cells 651 and 653. To ensure that the CPE of the subscriber destination 109 can reliably reassemble IP packets from a series of individual CCP cells, the CCP verifies that the pointer offset values and the previous PAP header's length field are in agreement. The first CCP cell 651 is followed by a subsequent CCP cell 653, each including respective CCP headers 655 and 657. The CCP cells 651 and 653 are not necessarily consecutive, in which case intermediate CCP cells include a CCP header with the maximum value. The first CCP cell 651 includes a PAP header 659 and a corresponding first portion of an IP packet 1. The CCP header 655 includes a pointer offset value indicating the position of the PAP header 659 within the CCP cell 651. The PAP header 659 includes the length field 621 defining the length of IP packet 1, and therefore indicates the location of a subsequent PAP header 661 within the subsequent CCP cell 653. The PAP header 661 is

located at the beginning of the next subsequent IP packet 2. The CCP header 657 includes a pointer offset value indicating the position of the PAP header 661 within the CCP cell 653. Thus, the CCP verifies that the CCP header 657 and the PAP header 659 are in agreement as to the location of the next PAP header 661.

Detailed Description Text (52):
Following the encoding process, a convolutional interleaving scheme is applied resulting in interleaved frames (not shown). In one embodiment, the resulting interleaved frames are composed of overlapping error-protected packets that are delimited by sync bytes to preserve a periodicity of 204 bytes. The frames may be interleaved in accordance with the ITU J.83 specification and will not be further described. The interleaved frames are then modulated, such as according to QAM-256 modulation as provided in the ITU J.83 specification. The QAM process adapts the synchronous, scrambled bit-stream for transmission over a channel as RF output. The QAM process blocks together bits from the data stream and then maps them into codewords using either Gray-codes or differential codes. The QAM process then converts the resulting digital codewords into an analog waveform based on a constellation diagram of combinations of amplitudes and phases, where each unique bit sequence corresponds to a point in the constellation.

Previous Doc Next Doc Go to Doc#

WEST Search History

Hide Items | Restore | Clear | Cancel

DATE: Tuesday, July 20, 2004

Hide?	Set Name	Query	Hit Count
		DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ	
☐	L17	20010817	2
☐	L16	20010817	95
☐	L15	L13 and l3	0
☐	L14	L13 and l11	0
☐	L13	(remove or removing or removal) near8 gap near8 (packet or stream)	139
☐	L12	l3 and l8	0
☐	L11	l3 and l7	2
☐	L10	l4 and l7	0
☐	L9	L8 and l7	0
☐	L8	handshake adj3 (logic or mechanism)	394
☐	L7	stream adj2 (map or mapping)	543
☐	L6	L4 and ll3	0
☐	L5	L4 and l3	0
☐	L4	byte near5 (packer or packing)	436
☐	L3	L2 and l1	133
☐	L2	(packet near8 (reassemble or reassembling))	1681
☐	L1	(packet near8 (split or splitting or splitter))	1915

END OF SEARCH HISTORY